

МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ

Учебно-методическое объединение по образованию
в области информатики и радиоэлектроники

УТВЕРЖДАЮ

Первый заместитель Министра образования
Республики Беларусь

_____ А.Г.Баханович

Регистрационный № ТД-_____/тип.

**ТЕХНОЛОГИИ ПРОЕКТИРОВАНИЯ ПРОГРАММНОГО
ОБЕСПЕЧЕНИЯ**

**Типовая учебная программа по учебной дисциплине
для специальности**

1-53 01 02 Автоматизированные системы обработки информации

СОГЛАСОВАНО

Председатель Учебно-методического
объединения по образованию в
области информатики и
радиоэлектроники

_____ В.А. Богуш

СОГЛАСОВАНО

Начальник Главного управления
профессионального образования
Министерства образования
Республики Беларусь

_____ С.Н.Пищов

СОГЛАСОВАНО

Проректор по научно-методической
работе Государственного учреждения
образования «Республиканский
институт высшей школы»

_____ И.В. Титович

Эксперт-нормоконтролер

Минск 2023

СОСТАВИТЕЛИ:

О.В.Герман, доцент кафедры информационных технологий автоматизированных систем учреждения образования «Белорусский государственный университет информатики и радиоэлектроники», кандидат технических наук, доцент;

Т.С.Боброва, старший преподаватель кафедры информационных технологий автоматизированных систем учреждения образования «Белорусский государственный университет информатики и радиоэлектроники»

РЕЦЕНЗЕНТЫ:

Кафедра «Программное обеспечение информационных систем и технологий» Белорусского национального технического университета (протокол № 7 от 22.02.2023);

С.Ф.Кондратюк, заместитель директора по работе с вузами и развитию персонала общества с ограниченной ответственностью «Софтарекс Технолоджиес»

РЕКОМЕНДОВАНА К УТВЕРЖДЕНИЮ В КАЧЕСТВЕ ТИПОВОЙ:

Кафедрой информационных технологий автоматизированных систем учреждения образования «Белорусский государственный университет информатики и радиоэлектроники» (протокол № 7 от 19.12.2022);

Научно-методическим советом учреждения образования «Белорусский государственный университет информатики и радиоэлектроники» (протокол № 7 от 17.03.2023);

Научно-методическим советом по разработке программного обеспечения и информационно-коммуникационным технологиям Учебно-методического объединения по образованию в области информатики и радиоэлектроники (протокол № 6 от 13.02.2023)

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

ХАРАКТЕРИСТИКА УЧЕБНОЙ ДИСЦИПЛИНЫ

Типовая учебная программа по учебной дисциплине «Технологии проектирования программного обеспечения» разработана для студентов учреждений высшего образования, обучающихся по специальности 1-53 01 02 Автоматизированные системы обработки информации в соответствии с требованиями образовательного стандарта высшего образования I ступени и типового учебного плана вышеуказанной специальности.

Знание и владение современными технологиями разработки программного обеспечения является важнейшей составляющей подготовки специалистов в области автоматизированных систем обработки информации. Учебная дисциплина «Технологии проектирования программного обеспечения» логически и методологически развивает общее направление объектно-ориентированного программирования и обеспечивает основу для применения его принципов при создании, оценке и эксплуатации программных систем различного назначения.

ЦЕЛЬ, ЗАДАЧИ УЧЕБНОЙ ДИСЦИПЛИНЫ

Цель учебной дисциплины: изучение и освоение современных технологий проектирования программного обеспечения на основе развития объектно-ориентированного подхода.

Задачи учебной дисциплины:

- изучение принципов объектно-ориентированного проектирования;
- приобретение знаний об основных концепциях и методах объектно-ориентированного моделирования;
- овладение методами и средствами проектирования и разработки программных систем на основе современных информационных технологий, языковых средств и паттернов (шаблонов) проектирования;
- изучение различных метрик и способов оценки качества программного обеспечения.

Базовыми учебными дисциплинами для учебной дисциплины «Технологии проектирования программного обеспечения» являются «Объектно-ориентированное программирование», «Основы алгоритмизации и программирования», «Построение и анализ алгоритмов». В свою очередь учебная дисциплина «Технологии проектирования программного обеспечения» является одной из базовых для таких учебных дисциплин, как «Мобильные приложения для информационных систем», «Современные системы программирования» (учебная дисциплина компонента учреждения высшего образования).

ТРЕБОВАНИЯ К УРОВНЮ ОСВОЕНИЯ СОДЕРЖАНИЯ УЧЕБНОЙ ДИСЦИПЛИНЫ

В результате изучения учебной дисциплины «Технологии проектирования программного обеспечения» формируются следующие компетенции:

универсальные:

владеть основами исследовательской деятельности, осуществлять поиск, анализ и синтез информации;

обладать навыками саморазвития и совершенствования в профессиональной деятельности;

проявлять инициативу и адаптироваться к изменениям в профессиональной деятельности;

базовая профессиональная: осуществлять объектный анализ и проектирование систем обработки информации.

В результате изучения учебной дисциплины студент должен:

знать:

основные понятия и концепции объектно-ориентированного проектирования и моделирования;

принципы структурного программирования и нисходящей разработки программ, современные модели разработки программного обеспечения;

паттерны объектно-ориентированного проектирования;

языковые средства для описания и построения моделей программ;

современные системы метрик и оценки качества программного обеспечения;

уметь:

применять на практике основные приемы эффективной разработки программ;

проектировать информационно-программное обеспечение с использованием современных информационных технологий, языковых средств и паттернов;

владеть:

навыками решения практических задач объектно-ориентированного проектирования и моделирования;

навыками работы в интегрированной среде проектирования программного обеспечения;

методами и приемами построения объектных моделей программных систем;

способами оценки качества программного обеспечения.

В рамках образовательного процесса по учебной дисциплине «Технологии проектирования программного обеспечения» студент должен приобрести не только теоретические и практические знания, умения и навыки по специальности, но и развить собственный ценностно-личностный и

интеллектуальный потенциал, формировать качества созидательной личности, готовой к активному участию в экономической, культурной и общественной жизни страны.

Типовая учебная программа рассчитана на 216 учебных часов, из них – 84 аудиторных. Примерное распределение аудиторных часов по видам занятий: лекции – 36 часов, лабораторные занятия – 32 часа, практические занятия – 16 часов.

ПРИМЕРНЫЙ ТЕМАТИЧЕСКИЙ ПЛАН

Наименование темы	Всего аудиторных часов	Лекции	Лабораторные занятия	Практические занятия
Введение. Эволюция языков программирования, жизненный цикл программ	2	2	-	-
Тема 1. Принципы структурного программирования и объектного проектирования	2	2	-	-
Тема 2. Современные методологии и модели разработки программного обеспечения: каскадная, спиральная, итеративная, V-модель и др.	2	2	-	-
Тема 3. Объектно-ориентированное моделирование на основе UML и иных модельных подходов	12	4	8	-
Тема 4. Проектирование программного обеспечения на основе функциональных моделей IDEF	6	2	4	-
Тема 5. Языковые средства спецификации программ и функций	10	2	4	4
Тема 6. Шаблоны (паттерны) проектирования	24	10	14	-
Тема 7. Оценка качества и программные метрики	10	4	-	6
Тема 8. Принципы верификации и тестирования программного обеспечения	10	4	2	4
Тема 9. Рефакторинг программного кода	4	2	-	2
Тема 10. Документация на разрабатываемое программное обеспечение	2	2	-	-
Итого:	84	36	32	16

СОДЕРЖАНИЕ УЧЕБНОГО МАТЕРИАЛА

ВВЕДЕНИЕ. ЭВОЛЮЦИЯ ЯЗЫКОВ ПРОГРАММИРОВАНИЯ, ЖИЗНЕННЫЙ ЦИКЛ ПРОГРАММ

Краткая история программирования, эволюция языков, основные технологические проблемы. Анализ жизненного цикла программы. Задачи проектирования программного обеспечения. Существующие системы проектирования и разработки программ.

Тема 1. ПРИНЦИПЫ СТРУКТУРНОГО ПРОГРАММИРОВАНИЯ И ОБЪЕКТНОГО ПРОЕКТИРОВАНИЯ

Нисходящее и восходящее программирование. Модульность. Пошаговая детализация. Структурные операторы. Технологии, основанные на структурном программировании (SADT и др.). Структурные принципы (в Java или другом языке высокого уровня). Принципы объектно-ориентированного проектирования для классов (принцип индивидуальной ответственности для класса, отделение интерфейса, принцип заменяемости и др.). Эвристические принципы проектирования.

Тема 2. СОВРЕМЕННЫЕ МЕТОДОЛОГИИ И МОДЕЛИ РАЗРАБОТКИ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ: КАСКАДНАЯ, СПИРАЛЬНАЯ, ИТЕРАТИВНАЯ, V-МОДЕЛЬ И ДР.

Сущность и анализ современных моделей разработки программного обеспечения: каскадной (водопадной), спиральной, итеративно-инкрементальной, V-модели, Agile, RAD-модели, «экстремальное программирование» и др. Их достоинства и недостатки. Функции персонала и распределение ролей коллектива разработчиков программного обеспечения. Современные CASE-системы.

Тема 3. ОБЪЕКТНО-ОРИЕНТИРОВАННОЕ МОДЕЛИРОВАНИЕ НА ОСНОВЕ UML И ИНЫХ МОДЕЛЬНЫХ ПОДХОДОВ

Диаграммы UML, сущность диаграммного подхода. Диаграмма вариантов использования. Примеры. Диаграммы классов. Атрибуты и методы классов. Виды зависимостей между классами. Диаграммы состояний и диаграммы деятельности. Связь между ними. Диаграммы компонентов и пакеты. Генерация каркаса приложения. Диаграммы кооперации. Диаграммы последовательности действий. Другие виды моделей: событийные сети, сети Петри и их различные типы, автоматы.

Тема 4. ПРОЕКТИРОВАНИЕ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ НА ОСНОВЕ ФУНКЦИОНАЛЬНЫХ МОДЕЛЕЙ IDEF

Сравнение объектного и функционального подходов к моделированию. Типы диаграмм IDEF – IDEF0, IDEF1X (сущность-связь), IDEF3, IDEF 4 и др. Требования по разработке. Принципы моделирования в IDEF0. Использование логических и синхронизирующих связей (условий) на диаграммах. Диаграммы

потоков данных – DFD. Создание функциональных диаграмм в интегрированной среде (например, BPWin). Примеры и объяснение. ER-диаграммы и нотация Чена-Баркера (модели IDEF1X). Переход от ER-диаграмм к логической модели данных.

Тема 5. ЯЗЫКОВЫЕ СРЕДСТВА СПЕЦИФИКАЦИИ ПРОГРАММ И ФУНКЦИЙ

Назначение спецификаций. Лямбда-выражения (Linq-конструкции) Рекурсивные спецификации и спецификации на основе правил. Объектный язык ограничений (OCL) в UML. Использование онтологических спецификаций в языках высокого уровня (типа Java, C++).

Тема 6. ШАБЛОНЫ (ПАТТЕРНЫ) ПРОЕКТИРОВАНИЯ

Понятие шаблона (паттерна) проектирования. Его назначение и использование. Классификация паттернов. Паттерн Строитель. Паттерн Синглетон. Паттерн Фасад. Паттерн Адаптер. Паттерн Декоратор. Паттерн Bridge (Мост). Паттерн Mediator (Посредник). Паттерн Стратегия. Паттерн Фабрика классов. Паттерн Абстрактная фабрика классов. Паттерн Итератор. Паттерн Состояние (State). Паттерн Команда (Command). Шаблон проектирования на основании MVC (Model-View-Controller). Примеры и объяснение реализаций на языке высокого уровня (Java, C++ и/или др.).

Тема 7. ОЦЕНКА КАЧЕСТВА И ПРОГРАММНЫЕ МЕТРИКИ

Требования к программному обеспечению. Классификация требований. Понятие качества программного обеспечения. Определение качества как степени (меры) удовлетворения требований. Использование метрических оценок (метрик) для оценки показателей качества. Виды метрик и их характеристики. Требования к надежности программ, модели надежности (Шумана, Шика-Волвертона и др.). Возможности измерения качественных показателей в современных системах программирования.

Тема 8. ПРИНЦИПЫ ВЕРИФИКАЦИИ И ТЕСТИРОВАНИЯ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

Определение верификации и тестирования. Отличие верификации от тестирования. Верификация структурных операторов присваивания, проверки условия, цикла, подпрограммы. Доказательство правильности программы. Модели тестирования: модель «черного ящика», «белого ящика», «серого ящика». Средства тестирования в современных языках программирования: оператор assert, создание unit-тестов, тестирование интерфейса на основе библиотеки Google Mock (или аналогов).

Тема 9. РЕФАКТОРИНГ ПРОГРАММНОГО КОДА

Понятие и назначение рефакторинга. Признаки плохого кода. Принципы и методы рефакторинга (изменение сигнатуры метода, перемещение метода, инкапсуляция поля, замена условного оператора полиморфным методом и др.).

Примеры. Проблемы, связанные с рефакторингом и способы их разрешения. Системы автоматизации рефакторинга.

Тема 10. ДОКУМЕНТАЦИЯ НА РАЗРАБАТЫВАЕМОЕ ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ

Стандарты на разрабатываемое программное обеспечение на примере ISO серии 9000, TickIT и др. Документация на программное обеспечение. Техническое задание, пояснительная записка, описание применения, тексты программ, программа методики испытаний и др. согласно ГОСТ 19.101. Содержание документов, разработка документации. Вопросы взаимодействия разработчика и заказчика. Основные проблемы и способы решения.

ИНФОРМАЦИОННО-МЕТОДИЧЕСКАЯ ЧАСТЬ**ЛИТЕРАТУРА****ОСНОВНАЯ**

1. Приемы объектно-ориентированного проектирования / Э. Гамма, Р. Хелм, Р. Джонсон [и др.] – Санкт-Петербург : Питер, 2015. – 368 с.
2. Мартин, Р. Чистый код: создание, анализ и рефакторинг / Р. Мартин. – Санкт-Петербург : Питер, 2019. – 464 с.
3. Коцюба, И. Ю. Основы проектирования информационных систем : учебное пособие / И. Ю. Коцюба, А. В. Чунаев, А. Н. Шиков. – Санкт-Петербург : Университет ИТМО, 2015. – 206 с.
4. Макконелл, С. Совершенный код / С. Макконелл. – Москва : Русская редакция, 2010. – 896 с.
5. Бураков, В. В. Управление качеством программных средств / В. В. Бураков. – Санкт-Петербург : СПбГУАП, 2009. – 287 с.

ДОПОЛНИТЕЛЬНАЯ

1. Фаулер, М. С++. Рефакторинг. Улучшение существующего кода / М. Фаулер. – Санкт-Петербург : Символ-Плюс, 2003. – 432 с.
2. Леоненков, А. В. Самоучитель UML / А. В. Леоненков. – Санкт-Петербург : Питер, 2004. – 430 с.
3. Макгрегор, Дж. Тестирование объектно-ориентированного программного обеспечения : практическое пособие / Дж. Макгрегор, Д. Сайкс. – Киев : ДияСофт, 2002. – 432 с.
4. Герман, Ю. О. Проектирование и разработка информационных систем : учебно-методическое пособие / Ю. О. Герман, О. В. Герман. – Минск : БГУИР, 2020. – 128 с.
5. Грис, Д. Наука программирования / Д. Грис. – Москва : Мир, 1984. – 416 с.
6. Куксенко, С. П. Современные технологии анализа и проектирования информационных систем / С. П. Куксенко. – Томск : ТГУ, 2016. – 101 с.
7. Коцюба, И. Ю. Методы оценки и измерения характеристик информационных систем : учебное пособие / И. Ю. Коцюба, А. В. Чунаев, А. Н. Шиков. – Санкт-Петербург : Университет ИТМО, 2015. – 264 с.
8. Куликов, С. Тестирование программного обеспечения. Базовый Курс / С. Куликов. – Минск : ЕРАМ-Systems, 2022. – 298с.

МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ ПО ОРГАНИЗАЦИИ И ВЫПОЛНЕНИЮ САМОСТОЯТЕЛЬНОЙ РАБОТЫ СТУДЕНТОВ

При изучении учебной дисциплины рекомендуется использовать следующие формы самостоятельной работы:

контролируемая самостоятельная работа в виде решения индивидуальных задач в аудитории во время проведения лабораторных занятий под контролем преподавателя.

ПЕРЕЧЕНЬ РЕКОМЕНДУЕМЫХ СРЕДСТВ ДИАГНОСТИКИ КОМПЕТЕНЦИЙ СТУДЕНТА

Типовым учебным планом по специальности 1-53 01 02 Автоматизированные системы обработки информации в качестве формы промежуточной аттестации по учебной дисциплине «Технологии проектирования программного обеспечения» рекомендуется экзамен и курсовой проект. Оценка учебных достижений студента производится по десятибалльной шкале.

Для текущего контроля по учебной дисциплине и диагностики компетенций студентов могут использоваться следующие формы:

- отчеты по лабораторным работам с их устной защитой;
- устный опрос на практических занятиях;
- электронные тесты;
- доклады на конференциях;
- оценивание на основе модульно-рейтинговой системы.

РЕКОМЕНДУЕМЫЕ МЕТОДЫ (ТЕХНОЛОГИИ) ОБУЧЕНИЯ

Основные рекомендуемые методы (технологии) обучения, отвечающие целям и задачам учебной дисциплины:

проблемное обучения (проблемное изложение, вариативное изложение, частично-поисковый метод), реализуемое на лекционных занятиях;

учебно-исследовательская деятельность, творческий подход, реализуемые на лабораторных и практических занятиях.

РЕКОМЕНДАЦИИ ПО КУРСОВОМУ ПРОЕКТИРОВАНИЮ

Курсовой проект по учебной дисциплине «Технологии проектирования программного обеспечения» представляет собой самостоятельную работу студента, выполняемую с целью с закрепления навыков практического программирования с использованием объектно-ориентированной парадигмы. Рекомендуемый объем пояснительной записки 25–30 страниц.

ПРИМЕРНЫЙ ПЕРЕЧЕНЬ ТЕМ КУРСОВЫХ ПРОЕКТОВ

1. Объектно-ориентированная модель информационной системы Интернет-магазина на основе UML.
2. Объектно-ориентированная модель информационной подсистемы для библиотеки на основе UML.
3. Объектно-ориентированная модель информационной подсистемы клиента электронной почты на основе UML.
4. Объектно-ориентированная модель информационной подсистемы «Трудоустройство» на основе UML.
5. Разработка модели программного обеспечения, для управления работой холодильника на основе UML.
6. Построение объектной модели предметной области «организация процессов спортивного клуба» с применением языка моделирования UML.
7. Информационная система для учета компьютерной техники на предприятии на основе UML.
8. Бизнес-модель на основе IDEF производства строительных материалов.
9. Бизнес-модель на основе IDEF производства кондитерских изделий.
10. Бизнес-модель на основе IDEF диагностики неисправностей и ремонта автомобиля.
11. Бизнес-модель на основе IDEF постройки дачного домика.
12. IDEF-модель технологии решения задачи линейного программирования и программный модуль на ее основе.
13. Программная система управления лифтом на основе шаблона MVC (Model-View-Controller).
14. Программная система управления бензозаправочной станцией на основе шаблона MVC (Model-View-Controller).
15. Программная система управления движением беспилотного транспортного средства на основе шаблона MVC (Model-View-Controller).
16. Программная система управления запасами на складе на основе шаблона MVC (Model-View-Controller).
17. Программная система выбора маршрута доставки товара на основе шаблона MVC (Model-View-Controller).
18. Реализация Unit-теста и программы размещения источников освещения в торговом помещении и оценка ее метрических показателей.
19. Реализация Unit-теста и программы оптимального размещения камер наблюдения в торговом помещении и оценка ее метрических показателей.
20. Реализация Unit-теста и программы размещения датчиков задымления в торговом помещении и оценка ее метрических показателей.
21. Разработка рекурсивной спецификации и реализация программы автоответчика по расписанию электропоездов.
22. Разработка рекурсивной спецификации и реализация программы автоответчика по заказу талона к врачу.

23. Разработка спецификации на основе правил и реализация программы для диагностики причин недомогания.

24. Разработка спецификации на основе правил и реализация программы для диагностики причин неисправности автомобиля.

25. Разработка спецификации на основе правил и реализация программы для контроля знаний обучаемого.

ПРИМЕРНЫЙ ПЕРЕЧЕНЬ ТЕМ ЛАБОРАТОРНЫХ РАБОТ

1. Моделирование (создание диаграмм) UML в среде Rational Rose.
2. Изучение и построение моделей линейки IDEF.
3. Лямбда-выражения (Linq-выражения) в языках высокого уровня.
4. Порождающие шаблоны (паттерны) проектирования.
5. Структурные шаблоны (паттерны) проектирования.
6. Поведенческие шаблоны (паттерны) проектирования.
7. Работа с шаблоном MVC (Model-View-Controller).
8. Использование Unit-тестов.

ПРИМЕРНЫЙ ПЕРЕЧЕНЬ ТЕМ ПРАКТИЧЕСКИХ ЗАНЯТИЙ

1. Разработка диаграмм вариантов использования UML.
2. Разработка рекурсивных спецификаций.
3. Создание программных спецификаций на основе правил.
4. Модели надежности программ.
5. Программные метрики и способы их вычислений.
6. Показатели качества программного обеспечения.
7. Проверка правильности кода с использованием нотации Хоара {предусловие} оператор {постусловие}.
8. Способы рефакторинга кода.

ПРИМЕРНЫЙ ПЕРЕЧЕНЬ КОМПЬЮТЕРНЫХ ПРОГРАММ

(необходимого оборудования, наглядных пособий и т. п.)

1. Программное обеспечение Microsoft Visual C++ (Visual Studio NET).
2. Программное обеспечение IBM Rational Rose, BPWin.